DOCUMENT RESUME

ED 174 194 IR 007 408

AUTHOR McWeeney, Mark G.
TITLE PASCAL Program to Compute Descriptive Statistics and Generate a Bar Graph for PSES (Purdue Self-Evaluation System for School Media Centers). Elementary Catalog.
PUB DATE Jul 78
NOTE 44p.; Some appendices may not reproduce well due to small, light type; For related documents, see ED 126 914-915 and IR 007 409

EDRS PRICE MF01 Plus Postage. PC Not Available from EDRS.
DESCRIPTORS Elementary Schools; *Instructional Materials Centers; Library Surveys; Questionnaires; *Statistical Analysis; Student Attitudes; Teacher Attitudes
IDENTIFIERS *PASCAL Program; Purdue Self Evaluation System

ABSTRACT

This report explains the output of a computer program which was developed for use by media specialists in calculating the statistics on data gathered from PSES questionnaires, as well as the computer program itself. Written in PASCAL, this program computes descriptive statistics, a bar graph, and percentages for the PSES instrument. Appendices provide data from three elementary schools where the PSES was tested, a flowchart of the computer program, and a copy of the PASCAL program. (RAO)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED174194

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

PASCAL Program to Compute Descriptive Statistics and Generate a Bar Graph for PSES (Purdue Self-Evaluation System for School Media Centers) Elementary Catalog

Mark G. McWeeney

Assistant Professor

University of Colorado - Boulder

July, 1978

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Mark G. McWeeney

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC) AND USERS OF THE ERIC SYSTEM."

© Copyright 1978, Mark G. McWeeney.
All rights reserved.

IR007408

Table of Contents

	Page
Abstract	iii
Chapter 1. Introduction	1
Purpose of Program	2
Administrator's Role	3
Chapter 2. Explanation of Program	4
PASCAL Output	4
Descriptive Statistics	4
Bar Graph	5
Percentages	6
Flowcharts	7
Chapter 3. PASCAL Program	8
Part 1. Table 1	8
Part 2. Figure 1	10
Part 3. Table 2	11
Appendices	
Appendix A: Table 1. Descriptive Statistics	12
Appendix B: Figure 1. Bar Graph	14
Appendix C: Table 2. Percentages	15
Appendix D: Tabulation Sheets	16
Appendix E: Flowcharts	18
Appendix F: PASCAL Program	22

Abstract

The PSES (Purdue Self-Evaluation System for School Media Centers, Elementary Catalog) was compiled in 1976 to enable the media specialist to evaluate his/her media center. Since media specialists are usually not experts in writing computer programs to calculate the statistics based on data tabulated from questionnaires, this author wrote a PASCAL program to compute descriptive statistics, a bar graph, and percentages for the PSES instrument.

This report explains the output the PASCAL program generates as well as the PASCAL program itself.

Chapter 1. Introduction

The PSES (Purdue Self-Evaluation System for School Media Centers) was compiled by David V. Loertscher and Janet G. Stroud from Purdue University in 1976 (ERIC Document: ED 126915). The purpose of the instrument is to provide the media specialist with a tool for evaluating his/her media center. Loertscher and Stroud tested their instrument in many of the elementary, middle, and senior high schcols in the State of Indiana. This author with his graduate students tested the PSES instrument in five schools (three, elementary; two, secondary) in the State of Colorado to 210 students, 170 elementary school students and 40 secondary school students.

The data provided in Appendixes A, B, and C are the result of cumulating the responses of students from three different elementary schools. In all three schools, the questionnaire was given to students in the 3rd grade (Group 1) and students in the 5th grade (Group 2).

Detailed results of each of the five schools participating in the study will be the subject of a future report, but this report explains the PASCAL program that compiles the statistics based on the data gathered from those schools who completed the PSES questionnaire.

Purpose of Program

Compiling all the statistics (mean of each item, the percentages of those responding Regularly, Occasionally, and Never for each item) is too time consuming for most media specialists. In fact, the thought of having to compile all the statistics deters the librarian from choosing any instrument which requires long hours spent in tabulating and compiling the results.

This situation led this author to write a program to 1. compile descriptive statistics, 2. generate a bar graph, and 3. estimate percentages. Not having to worry about compiling the results after tabulating the data allows the media specialist the time to administer the instrument.

There are two PSES instruments; one for the elementary school and another for the junior and senior high school. The PASCAL program contained in Appendix F will correctly compile the statistics for the elementary catalog only.

The difference between the program that calculates the statistics based on the PSES Elementary Catalog and the program that calculates the statistics based on the PSES Junior, Senior High School Catalog is not the number of items but the number of items selected under each of the eight service categories. Therefore, there are two PASCAL programs available; one to calculate results based on the Elementary Catalog, and the other on the Junior, High Catalog.

Administrator's Role

The media specialist must select fifty items from the 265 items from the PSES booklet or select fifty items from the student catalog. These items must then by typed, duplicated, and administered to those groups who have been chosen to respond to the questionnaire.

After the questionnaire is returned by the participants in the study, the tabulation procedure begins. A tabulation sheet (Appendix D) should be used for this task. Keypunching the data will also be easier if a tabulation sheet is used. Tabulation procedures include:

1. counting the number of those responding Regularly for Item 1, Group 1 and recording the number on the tabulation sheet;
2. counting the number of those responding Occasionally for Item 1, Group 1 and recording the number on the tabulation sheet; and
3. counting the number of those responding Rarely or Never for Item 1, Group 1 and recording the number on the tabulation sheet.

This procedure is repeated for each item in Group 1 and Group 2 until the sheet is completed.

The tabulation sheet becomes the source document for keypunching. Six numbers are typed on one computer card, the first three numbers for Item 1, Group 1, and the second three numbers for Item 1, Group 2. Then the next six numbers are keypunched on another card, three numbers for Item 2, Group 1; three numbers for Item 2, Group 2.

The procedure is repeated until all the data are keypunched. The data cards and the program cards are then submitted to the computing center for processing, using the PASCAL program to manipulate the data.

Chapter 2. Explanation of Program

The program can be explained in two ways: 1. the output the program generates and 2. the PASCAL program itself. Statistics performed on the data were based on a fifty item questionnaire. Fifty items were recommended by Loertscher and Stroud if the questionnaire were to be given to students. In the five case studies, each study was based on administering the questionnaire to students. The program was written to include eight services: Accessibility, Awareness, Professional, Utilization, Acquisition, Production, Evaluation, and Activity.

PASCAL Output

The PASCAL program generates three outputs: 1. Table 1, Descriptive Statistics (Appendix A); 2. Figure 1, Bar Graph (Appendix B); and 3. Table 2, Percentages (Appendix C). Table 1 produces descriptive statistics on two pages. The headings on the first sheet can be read in the following way:

Number--refers to one of the items selected from the PSES questionnaire;

Group 1 and Group 2--terms to distinguish those who responded; eg,

3rd Grade--5th Grade;

1--those responding Regularly;

2--those responding Occasionally;

3--those responding Rarely or Never:

M1--mean score response of Group 1;

M2--mean score response of Group 2;

D--difference. An asterisk will appear in this column if there is a 0.5 difference between M1 and M2;

SM1--Sum of the means of Group 1;

SM2--sum of the means of Group 2;

ASM1--the average of the sum of the means of Group 1;

ASM2--the average of the sum of the means of Group 2;

D--difference. An asterisk will appear in this column if a 0.5 difference occurs between ASM1 and ASM2.

The mean for each item is determined by weighing each item according to how the person responded to the item. Table 1 (Appendix A) shows that, for Number 1, Group 1, 73 people responded "Regularly". This number, 73, is multiplied by 1 giving 73. Seven (7) people responded "Occasionally." This number is multiplied by 2 giving 14. Two (2) people responded "Rarely or Never." This number is multiplied by 3 giving 6. These numbers: 73, 14, 6 equal 93. This number, 93, is divided by the total number responding to Item 1 which is 82 which gives a mean of 1.13. The mean for each item is determined in the same way.

Figure 1. Bar Graph

The second output the PASCAL program produces is a Bar Graph (Appendix B), which visually shows how each group responded for each service. The average of the sum of the means (ASM1 and ASM2) are used to graph the results. The headings serve to distinguish the range of the means. The asterisks and x's under "Services Regularly Performed" fall between the means 1.0 to 1.5. The asterisks and x's under "Services Occasionally Performed" fall between the means 1.5 to 2.5 with the means between 1.5 and 2.0 in the upper half. The asterisks and x's under "Services Rarely or Never Performed" fall between the means 2.5 and 3.0. The higher the mean, the poorer the service is being performed. Conversely, the lower the mean, the better the service is being performed.

Table 2. Percentages

The third output lists the percentages (Appendix C) for each item for both groups responding Regularly, Occasionally , or Never. The percentages for each item are determined by dividing the number of those responding Regularly, Occasionally, or Never by the total number responding to each item.

In Table 1 (Appendix A), seventy-three students responded Regularly to Item 1, Group 1. This number, 73, is divided by 82, the total number of students responding to this item. Thus, 89% responded Regularly to Item 1 from Group 1. Each percentage is determined in the same way.

Flowcharts

By definition, a flowchart is a graphical representation for the definition, analysis, or solution of a problem. The definition of the problem relates to what should be done with the data once the data have been tabulated. For the data collected from the questionnaire, there was a need to produce Table 1, Figure 1, and Table 2. The analysis of the problem relates to the method of producing the tables and figure; and the solution relates to the programming required to accomplish the task.

The flowcharts presented in Appendix E pertain to the analysis of the problem in that the method of producing Table 1, Figure 1, and Table 2 are diagrammed. The solution to the problem is presented in Appendix F, the programming required to accomplish the task.

The flowcharts diagram the steps which are required to produce each output. The functions are placed after each heading; for example, there are eleven symbols under the heading "First Output" (Appendix E, Flowchart 1). There are four categories of functions: processing ▭, decision ◇ , input-output ▱ , and procedure connector ○ .

The processing symbol states what has to be done (read a card, calculate means, percentages etc). The decision symbol is concerned with answering a question (Is there a difference in a mean?). The input-output symbol specifies that information must be printed on the line printer. The procedure connector informs the reader to look elsewhere for the flow of the procedure. The flow of the procedures is diagrammed on the last flowchart sheet.

Chapter 3. PASCAL Program

As stated previously, the program can be explained in two ways: 1. the output the program generates and 2. the program itself. This chapter will explain Number 2, the program itself. The program is written in PASCAL, a higher level programming language which can be learned by those who are not necessarily computer scientists, namely, teachers and media specialists.

The structure of this program has a main procedure starting on Page 6 and two other procedures: Procedure Writeout on Page 2 and Procedure Symbol on Page 2. Note: Appendix F has its own pagination printed on the top line of each page. Pages referred to in this section relate to the pages printed out by the computer.

The main procedure beginning on Page 6 is written in three parts with each part relating to one of the three outputs; namely, Table 1, Figure 1, and Table 2. Part 1 produces Table 1. After initializing the variables that were declared on Pages 1 and 2, the heading for Table 1 is printed for the first output. The data are then read in from the first data card (Read A1, B1, C1, A2, B2, C2) on Page 7. Six numbers are read in: 3 numbers corresponding to how students responded from Group 1 and 3 numbers corresponding to how students responded from Group 2. These numbers were keypunched onto cards using the Tabulation Sheet (Appendix D).

Statistics are then calculated (sum, mean, sum of means, and the total sum of the means) and percentages (Page 7) are calculated and written into an array called Per. These percentages are saved until they are needed when printing out Table 2. The statement, "If (Number = 7) then" on Page 7 tests whether

seven cards have been read in so far. Since only one card has been read in so far, the program moves down the series of statements to "End;" on Page 9 and returns to reading the numbers on the second card (Line 5 on Page 7).

When seven cards have been read in, the average sum of the mean for each group is calculated and rounded. The rounded mean is used for a Case statement later on in the program. A call to the procedure Writeout on Page 2 is now made. Procedure Writeout prints out the word Accessibility and an asterisk if there is a 0.5 difference between the mean of Group 1 and Group 2 under the column D on Table 1.

Seven cards refer to the number of items selected from the service, Accessibility, from the PSES booklet. How many items were selected from each group? The answer to this question is stated on Page 1 beginning on Line 4. The comment notes that this PASCAL program will work only if the following items from each group are selected. For example, there are 14 items in the student catalog for the service, Accessibility. Only seven items can be chosen from this group, not just the first seven items but any seven items. The seven items chosen can be in any order used in the questionnaire.

Each time a card is read in, a counter increases by one number and tests whether it has reached 7 (Accessibility), 14 (7 more items from Awareness) up to 50 (9 items from Activity). Throughout Part 1 of this program, after reading each card, the data on each card is printed out plus the statistics calculated as shown for each line from Table 1.

Note: If more than seven items are chosen under Accessibility, then the number of items chosen must be written into the program. The PASCAL program must be changed to accomodate specific needs. These changes must be made on Pages 7, 8, and 9. For example, if 10 items are chosen, then the statement, "If (Number = 7) then" must be changed to "If (Number = 10) then". Changing the number for one service will necessitate changing statements on Pages 7, 8, and 9. If five items are chosen under Awareness, then the statement on Page 8, First line, must be changed to "Else if (Number = 15) then." The user of this program is advised to ask a programmer to make the necessary changes throughout the program if changes are made which do not conform to the numbers chosen from each service as listed on Page 1 beginning on Line 4.

Part 2 produces Figure 1, the Bar Graph (Appendix B). This segment begins in the middle of Page 9 and continues through Page 11, Line 6. The headings are written out and, after each service is printed on the left of the bar graph, there is a call to the second Procedure, called Symbol on Page 2. The Procedure, Symbol, covers pages 2 through 6 and is built upon Case statements: "Case X of 10" through "Case X of 30."

These numbers, 10 through 30, are derived by rounding the average sum of the mean (ASM1) and multiplying by 10. For example, the ASM1 from Table 1 under Accessibility is 1.68. Multiplying this by 10 and rounding the number will yield 17. Thus Case X of 17 (Page 4) would be found in the Procedure Symbol, and the statements would be carried out one by one.

To summarize, the statements would distinguish between Group 1 and Group 2 and print out the appropriate number of *'s if Group 1 and the appropriate number of x's if Group 2. After each mean is multiplied and rounded, the Procedure Symbol would be called, and the statements would be carried out for each Case Statement. After the bar graph is printed, the Main procedure moves to Part 3.

Part 3 produces Table 2, Percentages (Appendix C) for each response for each item and for each group. Since the percentages were calculated when each card was processed and written into the array, the only thing now to do is to print them out. The program provides for this as specified on Page 11, Line 6, with the comment, "Write out Percentages." The statement which follows on Line 7, "Writeln('1')" is a carriage return command which ejects the paper to the next page prior to printing the output. The last statement, "END", ends the PASCAL program.

The purpose of this explanation was to give the reader a general idea how the program works; that is, the flow from Table 1 to Figure 1 to Table 2. The explanation just given was a verbal description of the flow of the program. The flowcharts present a visual representation of the flow of the program. After the flowchart is graphed to produce the output needed, the program is written in a language that would best produce what is needed. Since PASCAL could produce the desired output, the PASCAL language was chosen. However, the main reason for programming in PASCAL is that this author knows PASCAL better than other languages.

Appendix A

Table 1

Descriptive Statistics

NUMBER	GROUP 1			GROUP 2			M1	M2	D	SM1	SM2	ASM1	ASM2	D
	1	2	3	1	2	3								
1	73	7	2	71	12	5	1.13	1.25						
2	9	2	70	1	5	82	2.75	2.92						
3	55	15	12	44	40	4	1.48	1.5[illegible]						
4	21	19	42	16	29	40	2.26	2.28						
5	51	28	3	36	50	2	1.41	1.61						
6	72	9	1	47	35	6	1.13	1.53						
7	40	35	6	33	43	[illegible]	1.58	1.67						
								ACCESSIBILITY		11.75	12.82	1.68	1.83	
8	33	36	13	1[illegible]	52	20	1.76	2.05						
9	33	41	9	21	55	12	1.71	1.90						
10	33	33	25	11	1[illegible]	6[illegible]	1.91	2.58	*					
11	20	28	33	9	35	44	2.16	2.40						
12	40	36	5	31	4[illegible]	10	1.57	1.76						
13	41	32	10	29	[illegible]7	22	1.63	1.92						
14	26	40	13	19	44	25	1.84	2.07						
								AWARENESS		12.57	14.67	1.80	2.10	
15	41	31	10	30	40	18	1.62	1.86						
								PROFESSIONAL		1.62	1.86	1.62	1.86	
16	54	27	1	39	41	7	1.35	1.63						
17	32	32	8	17	69	13	1.67	1.96						
18	21	48	13	34	40	14	1.90	1.77						
19	27	39	16	18	[illegible]	19	1.87	2.01						
20	24	42	15	25	55	8	1.89	1.81						
21	44	35	3	48	34	5	1.50	1.51						
22	19	40	23	51	31	13	2.05	1.60						
								UTILIZATION		12.23	12.29	1.75	1.76	
23	22	27	33	4	27	57	2.13	2.60						
24	7	55	20	4	16	69	2.16	2.73	*					
25	20	25	37	10	20	56	2.21	2.53						
26	3	14	71	1	3	78	2.77	2.94						
27	13	56	19	12	27	41	2.07	2.36						
								ACQUISITIONS		11.34	13.17	2.27	2.63	

NUMBER	GROUP 1			GROUP 2			M1	M2	D	SM1	SM2	ASM1	ASM2	D
	1	2	3	1	2	3								
28	23	34	31	23	28	21	2.09	1.97						
29	30	44	14	31	40	11	1.82	1.76						
30	36	39	13	35	35	12	1.74	1.72						
31	45	29	14	39	28	15	1.65	1.71						
32	57	33	8	46	28	7	1.50	1.52						
33	11	49	28	18	34	30	2.19	2.15						
34	30	46	5	23	41	18	1.69	1.94						
								PRODUCTION		12.68	12.76	1.81	1.82	
35	46	32	10	73	9	1	1.59	1.13						
36	42	36	10	43	24	4	1.64	1.45						
37	39	40	8	50	32	0	1.64	1.39						
38	34	44	7	59	22	1	1.68	1.29						
39	26	49	13	25	47	10	1.85	1.82						
40	29	34	25	25	40	17	1.95	1.90						
41	37	28	22	30	38	11	1.83	1.76						
								EVALUATION		12.19	10.75	1.74	1.54	
42	66	21	1	73	8	1	1.26	1.12						
43	30	49	9	32	31	18	1.76	1.83						
44	44	34	10	37	39	6	1.61	1.62						
45	31	36	21	26	34	23	1.89	1.96						
46	43	33	12	45	32	5	1.65	1.51						
47	37	42	9	67	23	1	1.68	1.27						
48	32	38	8	39	38	4	1.69	1.57						
49	39	37	12	21	47	14	1.69	1.91						
50	37	31	20	42	32	8	1.81	1.59						
								ACTIVITY		15.04	14.39	1.67	1.60	
								TOTAL		89.42	92.70	1.79	1.85	

Table 1

Descriptive Statistics

Appendix B

Figure 1

Bar Graph

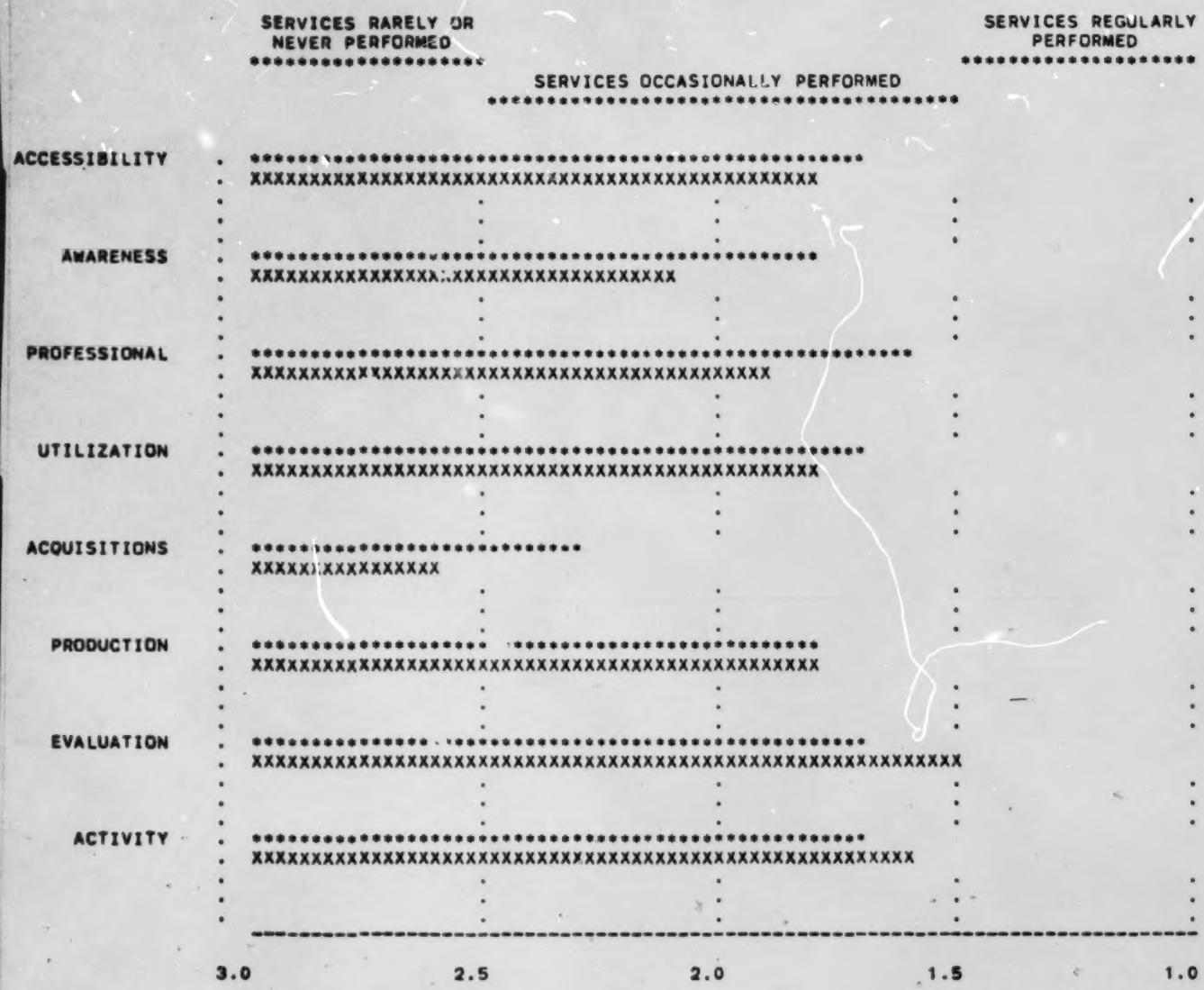


Figure 1

Bar Graph

Appendix C

Table 2

Percentages

PERCENTAGES

	GROUP 1			GROUP 2		
	1	2	3	1	2	3
	P	P	P	P	P	P
1.	89.	9.	2.	81.	14.	6.
2.	11.	2.	86.	1.	6.	93.
3.	67.	18.	15.	50.	45.	5.
4.	26.	23.	51.	19.	34.	47.
5.	62.	34.	4.	41.	57.	2.
6.	88.	11.	1.	53.	40.	7.
7.	49.	43.	7.	40.	52.	7.
8.	40.	44.	16.	18.	59.	23.
9.	40.	49.	11.	24.	62.	14.
10.	36.	36.	27.	12.	17.	70.
11.	25.	35.	41.	10.	40.	50.
12.	49.	45.	6.	36.	53.	11.
13.	49.	39.	12.	33.	42.	25.
14.	33.	51.	16.	22.	50.	28.
15.	50.	38.	12.	34.	45.	20.
16.	66.	33.	1.	45.	47.	8.
17.	44.	44.	11.	17.	70.	13.
18.	26.	59.	16.	39.	45.	16.
19.	33.	48.	20.	21.	57.	22.
20.	30.	52.	19.	28.	62.	9.
21.	54.	43.	4.	55.	39.	6.
22.	23.	49.	28.	54.	33.	14.
23.	27.	33.	40.	5.	31.	65.
24.	9.	67.	24.	4.	18.	78.
25.	24.	30.	45.	12.	23.	65.
26.	3.	16.	81.	1.	4.	95.
27.	15.	64.	22.	15.	34.	51.
28.	26.	39.	35.	32.	39.	29.
29.	34.	50.	16.	38.	49.	13.
30.	41.	44.	15.	43.	43.	15.
31.	51.	33.	16.	48.	34.	18.
32.	58.	34.	8.	57.	35.	9.
33.	12.	56.	32.	22.	41.	37.
34.	37.	57.	6.	28.	50.	22.
35.	52.	36.	11.	88.	11.	1.
36.	48.	41.	11.	61.	34.	6.
37.	45.	46.	9.	61.	39.	0
38.	40.	52.	8.	72.	27.	1.
39.	30.	56.	15.	30.	57.	12.
40.	33.	39.	28.	30.	49.	21.
41.	43.	32.	25.	38.	48.	14.
42.	75.	24.	1.	89.	10.	1.
43.	34.	56.	10.	40.	38.	22.
44.	50.	39.	11.	45.	48.	7.
45.	35.	41.	24.	31.	41.	28.
46.	49.	37.	14.	55.	39.	6.
47.	42.	48.	10.	74.	25.	1.
48.	41.	49.	10.	48.	47.	5.
49.	44.	42.	14.	26.	57.	17.
50.	42.	35.	23.	51.	39.	10.

Table 2

Percentages

Appendix D

Tabulation Sheets

TABULATION SHEET

	Group 1			Group 2		
	R	O	N	R	O	N
1.						
2.						
3.						
4.						
5.						
6.						
7.						
8.						
9.						
10.						
11.						
12.						
13.						
14.						
15.						
16.						
17.						
18.						
19.						
20.						
21.						
22.						
23.						
24.						
25.						

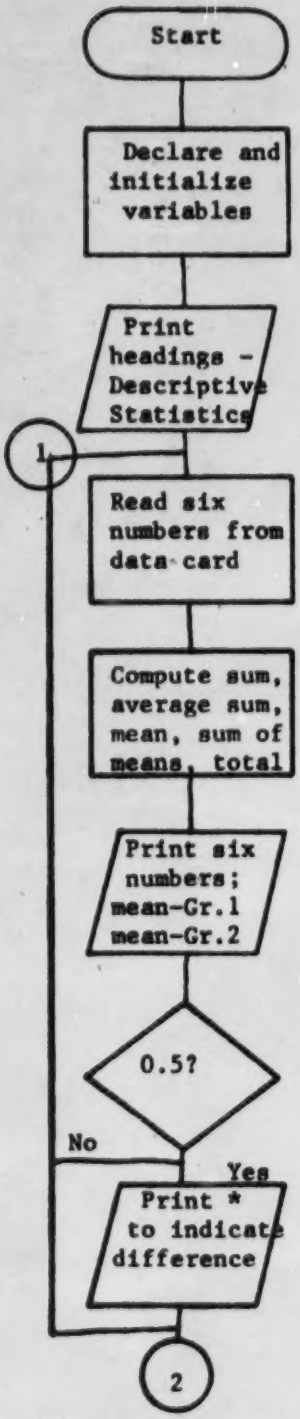
TABULATION SHEET

	Group 1			Group 2		
	R	O	N	R	O	N
26.	____	____	____	____	____	____
27.	____	____	____	____	____	____
28.	____	____	____	____	____	____
29.	____	____	____	____	____	____
30.	____	____	____	____	____	____
31.	____	____	____	____	____	____
32.	____	____	____	____	____	____
33.	____	____	____	____	____	____
34.	____	____	____	____	____	____
35.	____	____	____	____	____	____
36.	____	____	____	____	____	____
37.	____	____	____	____	____	____
38.	____	____	____	____	____	____
39.	____	____	____	____	____	____
40.	____	____	____	____	____	____
41.	____	____	____	____	____	____
42.	____	____	____	____	____	____
43.	____	____	____	____	____	____
44.	____	____	____	____	____	____
45.	____	____	____	____	____	____
46.	____	____	____	____	____	____
47.	____	____	____	____	____	____
48.	____	____	____	____	____	____
49.	____	____	____	____	____	____
50.	____	____	____	____	____	____

Appendix E

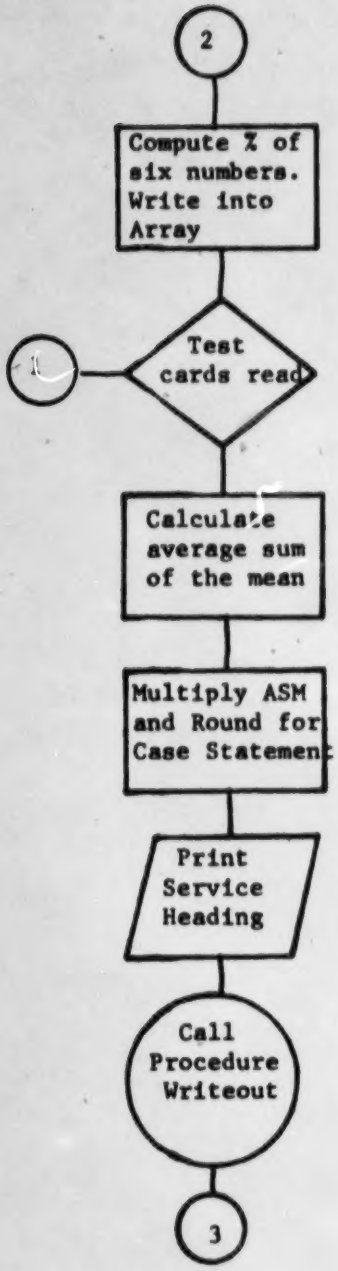
Flowcharts

Flowchart 1



First Output

Test for 0.5 difference between Mean 1 and Mean 2. If different proceed.



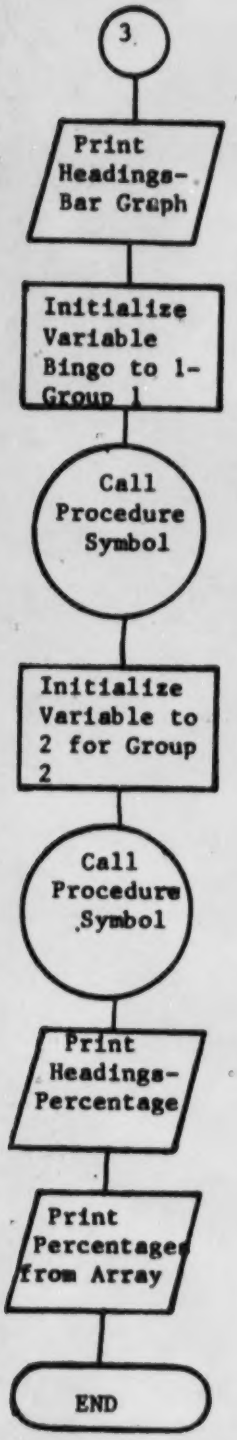
Flowchart 2

If number of cards = 7, 14, 15, 22, 27, 34, 41, or 50, proceed; else loop to Read statement

The average sum of the mean will fall between 1.0 and 3.0

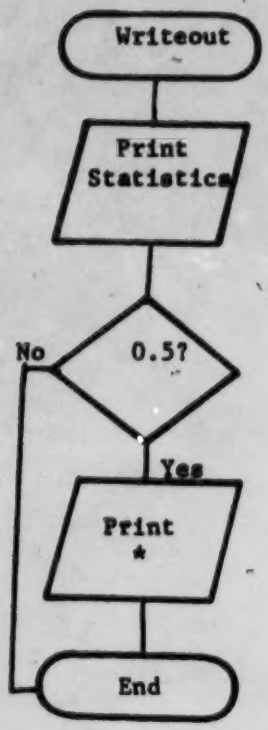
Once rounded, the ASM will be between 10 and 30; 21 numbers

Flowchart 3



Second Output

Third Output

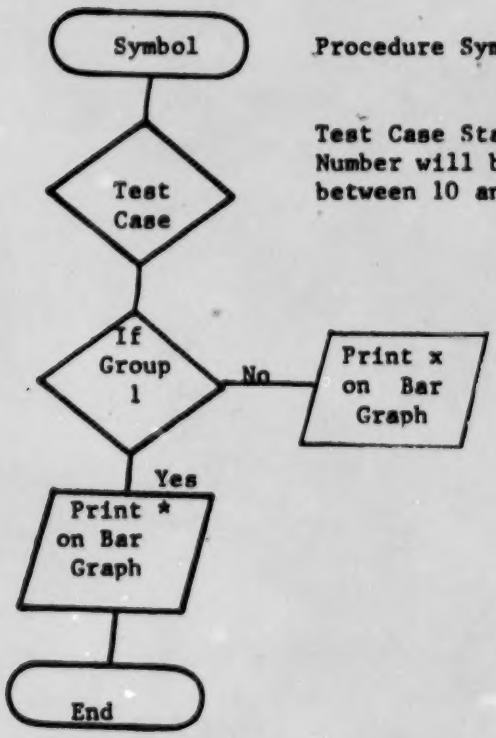


Procedure Writeout

Flowchart 4

Print sum of mean for each group and the average sum of mean for each group

Is there a 0.5 difference between ASM1 and ASM2?



Procedure Symbol

Test Case Statement Number will be between 10 and 30

Appendix F

PASCAL Program

```
00006  0 (* THIS PROGRAM COMPILES THE STATISTICS FOR LOERTSCHER'S AND STROUD'S PSES,
00006  0 ELEMENTARY SCHOOL CATALOG.  THIS IS NOT AN OP-SCAN PROGRAM, FOR IT
00006  0 DOES NOT TABULATE THE NUMBER OF RESPONSES TO EACH QUESTION.
00006  0
00006  0 FOR THIS PROGRAM TO WORK CORRECTLY, THERE MUST BE THE FOLLOWING:
00006  0      1.  7 ITEMS FROM ACCESSIBILITY,
00006  0      2.  7 ITEMS FROM AWARENESS,
00006  0      3.  1 ITEM FROM PROFESSIONAL,
00006  0      4.  7 ITEMS FROM UTILIZATION,
00006  0      5.  5 ITEMS FROM ACQUISITIONS,
00006  0      6.  7 ITEMS FROM PRODUCTION,
00006  0      7.  7 ITEMS FROM EVALUATION, AND
00006  0      8.  9 ITEMS FROM ACTIVITY.
00006  0 THERE MUST BE 50 ITEMS.  NO ITEMS WERE CHOSEN FROM PLANNING.
00006  0
00006  0 THE BEST USE OF PSES FOR THIS PROGRAM TO BE BENEFICIAL IS THE COMPARING
00006  0 OF TWO STUDENT GROUP RESPONSES.
00006  0
00006  0 THE PERSON USING THIS PROGRAM MUST TABULATE THE NUMBER OF RESPONSES UNDER
00006  0 1. REGULARLY, 2. OCCASIONALLY, AND 3. RARELY FOR EACH GROUP.  KEYPUNCH 6
00006  0 NUMBERS FOR EACH OF THE 50 ITEMS. *)
00006  0 PROGRAM ELEM(INPUT,OUTPUT);
04064  0 CONST CARDS = 50;
04064  0       UNDER = '----------------------------------------';
04064  0       UNDE  = '----------------------------------------';
04064  0       ZERO = 00.0;
04064  0 VAR A1,        (* NUMBER RESPONDING REGULARLY IN GROUP 1 *)
04064  0     B1,        (* NUMBER RESPONDING OCCASIONALLY IN GROUP 1 *)
04064  0     C1,        (* NUMBER RESPONDING NEVER IN GROUP 1 *)
04064  0     A2,        (* NUMBER RESPONDING REGULARLY IN GROUP 1 *)
04064  0     B2,        (* NUMBER RESPONDING OCCASIONALLY IN GROUP 2 *)
04064  0     C2 :       (* NUMBER RESPONDING NEVER IN GROUP 2 *)
04064  0          INTEGER;
04072  0     M1,        (* MEAN OF GROUP 1 *)
04072  0     M2 :       (* MEAN OF GROUP 2 *)
04072  0          REAL;
04074  0     SUM1,      (* SUM OF RESPONSES OF GROUP 1 *)
04074  0     SUM2,      (* SUM OF RESPONSES OF GROUP 2 *)
04074  0     ASUM1,     (* AVERAGE OF SUM1 *)
04074  0     ASUM2 :    (* AVERAGE OF SUM2 *)
04074  0          INTEGER;
04100  0     SKIP,      (* VARIABLE USED TO SKIP LINES *)
04100  0     I,         (* INDEX VARIABLE *)
04100  0     NUMBER :   (* VARIABLE USED TO COUNT CARDS *)
04100  0          INTEGER;
04103  0     SM1,       (* SUM OF THE MEANS OF GROUP 1 *)
04103  0     SM2,       (* SUM OF THE MEANS OF GROUP 2 *)
04103  0     ASM1,      (* AVERAGE OF SM1 *)
04103  0     ASM2,      (* AVERAGE OF SM2 *)
04103  0     TSM1,      (* TOTAL SUM OF THE MEANS OF GROUP 1 *)
04103  0     TSM2,      (* TOTAL SUM OF THE MEANS OF GROUP 2 *)
04103  0     ATSM1,     (* AVERAGE OF THE TOTAL OF THE SUM OF M1 *)
04103  0     ATSM2 :    (* AVERAGE OF THE TOTAL OF THE SUM OF M2 *)
04103  0          REAL;
04113  0     ACCES1,
04113  0     ACCES2,
04113  0     AWARE1,
```

```
04113  0      AWARE2,
04113  0      PROF1,
04113  0      PROF2,
04113  0      UTIL1,
04113  0      UTIL2,
04113  0      ACQUI1,
04113  0      ACQUI2,
04113  0      PROD1,
04113  0      PROD2,
04113  0      EVAL1,
04113  0      EVAL2,
04113  0      ACTIV1,
04113  0      ACTIV2 : INTEGER;
04133  0      X : INTEGER;
04134  0      BINGO : INTEGER;
04135  0
04135  0      PER : ARRAY[1..50,1..6] OF REAL;
04611  0      PA1,        (* PERCENT OF A1 *)
04611  0      PB1,        (* PERCENT OF B1 *)
04611  0      PC1,        (* PERCENT OF C1 *)
04611  0      PA2,        (* PERCENT OF A2 *)
04611  0      PB2,        (* PERCENT OF B2 *)
04611  0      PC2 :       (* PERCENT OF C2 *)
04611  0            REAL;
04617  0      J : INTEGER;
04620  0
04620  0
04620  0 PROCEDURE WRITEOUT;
00003  0   BEGIN
00003  1      WRITE(SM1:10:2,SM2:10:2,ASM1:10:2,ASM2:10:2);
00031  1 IF (ASM1 - ASM2 > 0.5) OR (ASM2 - ASM1 > 0.5)
00034  1      THEN WRITE("*":5);
00040  1      WRITELN;
00041  1      SM1 := 0;
00044  1      SM2 := 0;
00047  1      FOR SKIP := 1 TO 3 DO
00047  1      WRITELN(" ");
00056  1   END;
00062  0
00062  0 PROCEDURE SYMBOL(VAR X : INTEGER);
00004  0 BEGIN
00004  1   CASE X OF
00013  2      10 :
00013  2        BEGIN
00013  3        FOR I := 1 TO 80 DO
00014  3        BEGIN
00016  4        IF BINGO = 1 THEN
00017  4        WRITE("*")
00021  4        ELSE WRITE("X");
00024  4        END;
00026  3        WRITELN;
00027  3        END;
00030  2
00030  2      11 :
00030  2        BEGIN
00030  3        FOR I := 1 TO 76 DO
00031  3        BEGIN
```

```
00033  4        IF BINGO = 1 THEN
00034  4        WRITE("*")
00036  4        ELSE WRITE("X");
00041  4        END;
00043  3        WRITELN;
00044  3        END;
00045  2
00045  2      12 :
00045  2        BEGIN
00045  3        FOR I := 1 TO 72 DO
00048  3        BEGIN
00050  4        IF BINGO = 1 THEN
00051  4        WRITE("*")
00053  4        ELSE WRITE("X");
00056  4        END;
00060  3        WRITELN;
00061  3        END;
00062  2
00062  2      13 :
00062  2        BEGIN
00062  3        FOR I := 1 TO 68 DO
00063  3        BEGIN
00065  4        IF BINGO = 1 THEN
00066  4        WRITE("*")
00070  4        ELSE WRITE("X");
00073  4        END;
00075  3        WRITELN;
00076  3        END;
00077  2
00077  2      14 :
00077  2        BEGIN
00077  3        FOR I := 1 TO 64 DO
00100  3        BEGIN
00102  4        IF BINGO = 1 THEN
00103  4        WRITE("*")
00105  4        ELSE WRITE("X");
00110  4        END;
00112  3        WRITELN;
00113  3        END;
00114  2
00114  2      15 :
00114  2        BEGIN
00114  3        FOR I := 1 TO 60 DO
00115  3        BEGIN
00117  4        IF BINGO = 1 THEN
00120  4        WRITE("*")
00122  4        ELSE WRITE("X");
00125  4        END;
00127  3        WRITELN;
00130  3        END;
00131  2
00131  2      16 :
00131  2        BEGIN
00131  3        FOR I := 1 TO 56 DO
00132  3        BEGIN
00134  4        IF BINGO = 1 THEN
00135  4        WRITE("*")
```

```
00137  4        ELSE WRITE("X");
00142  4        END;
00144  3        WRITELN;
00145  3        END;
00146  2
00146  2      17 :
00146  2        BEGIN
00146  3        FOR I := 1 TO 52 DO
00147  3        BEGIN
00151  4        IF BINGO = 1 THEN
00152  4        WRITE("*")
00154  4        ELSE WRITE("X");
00157  4        END;
00161  3        WRITELN;
00162  3        END;
00163  2
00163  2      18 :
00163  2        BEGIN
00163  3        FOR I := 1 TO 48 DO
00164  3        BEGIN
00166  4        IF BINGO = 1 THEN
00167  4        WRITE("*")
00171  4        ELSE WRITE("X");
00174  4        END;
00176  3        WRITELN;
00177  3        END;
00200  2
00200  2      19 :
00200  2        BEGIN
00200  3        BEGIN
00200  4        FOR I := 1 TO 44 DO
00201  4        IF BINGO = 1 THEN
00204  4        WRITE("*")
00208  4        ELSE WRITE("X");
00213  4        END;
00213  3        WRITELN;
00214  3        END;
00215  2
00215  2      20 :
00215  2        BEGIN
00215  3        FOR I := 1 TO 40 DO
00216  3        BEGIN
00220  4        IF BINGO = 1 THEN
00221  4        WRITE("*")
00223  4        ELSE WRITE("X");
00226  4        END;
00230  3        WRITELN;
00231  3        END;
00232  2
00232  2      21 :
00232  2        BEGIN
00232  3        FOR I := 1 TO 36 DO
00233  3        BEGIN
00235  4        IF BINGO = 1 THEN
00236  4        WRITE("*")
00240  4        ELSE WRITE("X");
00243  4        END;
```

```
00245  3        WRITELN;
00246  3        END;
00247  2
00247  2      22 :
00247  2        BEGIN
00247  3        FOR I := 1 TO 32 DO
00250  3        BEGIN
00252  4        IF BINGO = 1 THEN
00253  4        WRITE("*")
00255  4        ELSE WRITE("X");
00260  4        END;
00262  3        WRITELN;
00263  3        END;
00264  2
00264  2      23 :
00264  2        BEGIN
00264  3        FOR I := 1 TO 28 DO
00265  3        BEGIN
00267  4        IF BINGO = 1 THEN
00270  4        WRITE("*")
00272  4        ELSE WRITE("X");
00275  4        END;
00277  3        WRITELN;
00300  3        END;
00301  2
00301  2      24 :
00301  2        BEGIN
00301  3        FOR I := 1 TO 24 DO
00302  3        BEGIN
00304  4        IF BINGO = 1 THEN
00305  4        WRITE("*")
00307  4        ELSE WRITE("X");
00312  4        END;
00314  3        WRITELN;
00315  3        END;
00316  2
00316  2      25 :
00316  2        BEGIN
00316  3        FOR I := 1 TO 20 DO
00317  3        BEGIN
00321  4        IF BINGO = 1 THEN
00322  4        WRITE("*")
00324  4        ELSE WRITE("X");
00327  4        END;
00331  3        WRITELN;
00332  3        END;
00333  2
00333  2      26 :
00333  2        BEGIN
00333  3        FOR I := 1 TO 16 DO
00334  3        BEGIN
00336  4        IF BINGO = 1 THEN
00337  4        WRITE("*")
00341  4        ELSE WRITE("X");
00344  4        END;
00346  3        WRITELN;
00347  3        END;
```

```
00350  2
00350  2      27 :
00350  2        BEGIN
00350  3        FOR I := 1 TO 12 DO
00351  3        BEGIN
00353  4        IF BINGO = 1 THEN
00354  4        WRITE("*")
00356  4        ELSE WRITE("X");
00361  4        END;
00363  3        WRITELN;
00364  3        END;
00365  2
00365  2      28 :
00365  2        BEGIN
00365  3        FOR I := 1 TO 8 DO
00366  3        BEGIN
00370  4        IF BINGO = 1 THEN
00371  4        WRITE("*")
00373  4        ELSE WRITE("X");
00376  4        END;
00400  3        WRITELN;
00401  3        END;
00402  2
00402  2      29 :
00402  2        BEGIN
00402  3        FOR I := 1 TO 4 DO
00403  3        BEGIN
00405  4        IF BINGO = 1 THEN
00406  4        WRITE("*")
00410  4        ELSE WRITE("X");
00413  4        END;
00415  3        WRITELN;
00416  3        END;
00417  2
00417  2      30 :
00417  2        BEGIN
00417  3        IF BINGO = 1 THEN
00421  3        WRITE("*")
00423  3        ELSE WRITE("X");
00426  3         WRITELN;
00427  3        END;
00430  2
00430  2    END; (* END CASE STATEMENT *)
00455  1    END; (* END PROCEDURE *)
00462  0
00462  0 BEGIN (* PROGRAM STARTS HERE *)
00462  1    SM1 := 0;
00027  1    SM2 := 0;
00032  1    TSM1 := 0;
00034  1    TSM2 := 0;
00037  1    NUMBER := 0;
00037  1    WRITELN("1");
00042  1    FOR SKIP := 1 TO 5 DO
00043  1    WRITELN(" ");
00052  1    WRITELN(" ","NUMBER":10,"GROUP 1":19,"GROUP 2":20,"M1":11,"M2":10,
00105  1    "D":5,"SM1":10,"SM2":10,"ASM1":10,"ASM2":10,"D":5);
00142  1    WRITELN(" "," ":22,"1  2  3"," ":13,"1  2  3");
```

```
00167  1   WRITELN;
00170  1
00170  1   FOR I := 1 TO CARDS DO
00171  1   BEGIN
00173  2     NUMBER := NUMBER + 1;
00174  2   READ(A1, B1, C1, A2, B2, C2);
00216  2   SUM1 := A1 + B1 + C1;
00220  2     SUM2 := A2 + B2 + C2;
00222  2     ASUM1 := A1 * 1  +  B1 * 2  +  C1 * 3;
00224  2     ASUM2 := A2 * 1  +  B2 * 2  +  C2 * 3;
00226  2     M1 := ASUM1 / SUM1;
00233  2     M2 := ASUM2 / SUM2;
00240  2     SM1 := SM1 + M1;
00242  2     SM2 := SM2 + M2;
00243  2     TSM1 := TSM1 + M1;
00244  2     TSM2 := TSM2 + M2;
00245  2     WRITE(" ",NUMBER:8,A1:15,B1:3,C1:3,A2:14,B2:3,C2:3,M1:11:2,M2:10:2);
00315  2     IF(M1 - M2 > 0.5) OR (M2 - M1 > 0.5) THEN
00321  2     WRITE("*":5);
00324  2     WRITELN;
00325  2
00325  2   (* THE FOLLOWING IS USED TO DETERMINE THE PERCENTAGE OF THOSE
00325  2   RESPONDING REGULARLY, OCCASIONALLY, OR NEVER *)
00325  2   PA1 := A1 * 100 / SUM1;
00335  2   PB1 := B1 * 100 / SUM1;
00342  2   PC1 := C1 * 100 / SUM1;
00350  2   PA2 := A2 * 100 / SUM2;
00356  2   PB2 := B2 * 100 / SUM2;
00364  2   PC2 := C2 * 100 / SUM2;
00372  2
00372  2   (* THESE PERCENTAGES ARE WRITTEN INTO THE ARRAY PER *)
00372  2   J := J + 1;
00374  2   PER[J,1] := PA1;
00400  2     IF PA1 = 0 THEN PER[J,1] := ZERO;
00410  2   PER[J,2] := PB1;
00416  2     IF PB1 = 0 THEN PER[J,2] := ZERO;
00425  2   PER[J,3] := PC1;
00433  2     IF PC1 = 0 THEN PER[J,3] := ZERO;
00442  2   PER[J,4] := PA2;
00450  2     IF PA2 = 0 THEN PER[J,4] := ZERO;
00457  2   PER[J,5] := PB2;
00465  2     IF PB2 = 0 THEN PER[J,5] := ZERO;
00474  2   PER[J,6] := PC2;
00502  2     IF PC2 = 0 THEN PER[J,6] := ZERO;
00511  2   (* END WRITING IN THE PERCENTAGES *)
00511  2
00511  2
00511  2     IF (NUMBER = 7) THEN
00513  2     BEGIN
00513  3     ASM1 := SM1 / 7;
00517  3     ASM2 := SM2 / 7;
00522  3     WRITE(" ","ACCESSIBILITY":75);
00531  3   ACCES1 := ROUND(ASM1 * 10);
00537  3   ACCES2 := ROUND(ASM2 * 10);
00544  3     WRITEOUT;
00545  3     END
00545  2
```

```
00545  2      ELSE IF (NUMBER = 14) THEN
00550  2      BEGIN
00550  3      ASM1 := SM1 / 7;
00553  3      ASM2 := SM2 / 7;
00556  3      WRITE(' ',"AWARENESS":75);
00565  3    AWARE1 := ROUND(ASM1 * 10);
00573  3    AWARE2 := ROUND(ASM2 * 10);
00600  3      WRITEOUT;
00601  3      END
00601  2
00601  2      ELSE IF (NUMBER = 15) THEN
00604  2      BEGIN
00604  3      ASM1 := SM1 / 1;
00607  3      ASM2 := SM2 / 1;
00612  3      WRITE(" ","PROFESSIONAL":75);
00621  3    PROF1 := ROUND(ASM1 * 10);
00627  3    PROF2 := ROUND(ASM2 * 10);
00634  3      WRITEOUT;
00635  3      END
00635  2
00635  2      ELSE IF (NUMBER = 22) THEN
00640  2      BEGIN
00640  3      ASM1 := SM1 / 7;
00643  3      ASM2 := SM2 / 7;
00646  3      WRITE(" ","UTILIZATION":75);
00655  3    UTIL1 := ROUND(ASM1 * 10);
00663  3    UTIL2 := ROUND(ASM2 * 10);
00670  3      WRITEOUT;
00671  3      END
00671  2
00671  2      ELSE IF (NUMBER = 27) THEN
00674  2      BEGIN
00674  3      ASM1 := SM1 / 5;
00700  3      ASM2 := SM2 / 5;
00703  3      WRITE(" ","ACQUISITIONS":75);
00712  3    ACQUI1 := ROUND(ASM1 * 10);
00720  3    ACQUI2 := ROUND(ASM2 * 10);
00725  3      WRITEOUT;
00726  3    WRITELN("1");
00731  3    FOR SKIP := 1 TO 5 DO
00732  3    WRITELN(' ");
00741  3    WRITELN(" ","NUMBER":10,"GROUP 1":19,"GROUP 2":20,"M1":11,"M2":10,
00774  3    "D":5,"SM1":10,"SM2":10,"ASM1":10,"ASM2":10,"D":5);
01031  3    WRITELN(" "," ":22,"1  2  3"," ":13,"1  2  3");
01056  3    WRITELN;
01057  3      END
01057  2
01057  2      ELSE IF (NUMBER = 34) THEN
01062  2      BEGIN
01062  3      ASM1 := SM1 / 7;
01065  3      ASM2 := SM2 / 7;
01070  3      WRITE(" ","PRODUCTION":75);
01077  3    PROD1 := ROUND(ASM1 * 10);
01105  3    PROD2 := ROUND(ASM2 * 10);
01112  3      WRITEOUT;
01113  3      END
01113  2
```

```
01113  2      ELSE IF (NUMBER = 41) THEN
01116  2      BEGIN
01116  3      ASM1 := SM1 / 7;
01122  3      ASM2 := SM2 / 7;
01125  3      WRITE(" ","EVALUATION":75);
01134  3    EVAL1 := ROUND(ASM1 * 10);
01142  3    EVAL2 := ROUND(ASM2 * 10);
01147  3      WRITEOUT;
01150  3      END
01150  2
01150  2      ELSE IF (NUMBER = 50) THEN
01153  2      BEGIN
01153  3      ASM1 := SM1 / 9;
01157  3      ASM2 := SM2 / 9;
01162  3      WRITE(" ","ACTIVITY":75);
01171  3    ACTIV1 := ROUND(ASM1 * 10);
01177  3    ACTIV2 := ROUND(ASM2 * 10);
01204  3      WRITEOUT;
01205  3      END;
01205  2    END;
01207  1    WRITELN;
01210  1    ATSM1 := TSM1 / 50;
01214  1    ATSM2 := TSM2 / 50;
01217  1    WRITE(" ","TOTAL":75,TSM1:10:2,TSM2:10:2,ATSM1:10:2,ATSM2:10:2);
01252  1 IF (ATSM1 - ATSM2 > 0.5) OR (ATSM2 - ATSM1 > 0.5)
01255  1    THEN WRITE("*":5);
01261  1    WRITELN;
01262  1
01262  1 WRITELN("1");
01265  1
01265  1    FOR SKIP := 1 TO 2 DO
01266  1    WRITELN(" ");
01275  1    WRITELN(" "," SERVICES RARELY OR ":47," SERVICES REGULARLY ":61);
01312  1    WRITELN(" ","  NEVER PERFORMED   ":47,"      PERFORMED      ":61);
01327  1    WRITELN(" ","********************":47,"********************":60);
01344  1    WRITELN("+","********************":47,"********************":60);
01361  1    WRITELN(" ","      SERVICES OCCASIONALLY PERFORMED      ":87);
01371  1    WRITELN(" ","*****************************************":87);
01401  1    WRITELN("+","*****************************************":87);
01411  1    WRITELN("0");
01414  1    WRITE(" ","ACCESSIBILITY":20,".":5,"  ");
01434  1    BINGO := 1;
01435  1    SYMBOL(ACCES1);
01437  1    BINGO := 2;
01440  1    WRITE(" ",".":25,"  ");
01453  1    SYMBOL(ACCES2);
01455  1
01455  1    FOR I := 1 TO 3 DO
01456  1    WRITELN(" ",".":25,".":22,".":20,".":20,".":20);
01511  1    WRITE(" ","AWARENESS":20,".":5,"  ");
01531  1    BINGO := 1;
01532  1    SYMBOL(AWARE1);
01534  1    BINGO := 2;
01535  1    WRITE(" ",".":25,"  ");
01550  1    SYMBOL(AWARE2);
01552  1
01552  1    FOR I := 1 TO 3 DO
```

```
01553  1   WRITELN(" ",".":25,".":22,".":20,".":20,".":20);
01606  1   WRITE(" ","PROFESSIONAL":20,".":5,"  ");
01626  1   BINGO := 1;
01627  1   SYMBOL(PROF1);
01631  1   BINGO := 2;
01632  1   WRITE(" ",".":25,"  ");
01645  1   SYMBOL(PROF2);
01647  1
01647  1   FOR I := 1 TO 3 DO
01650  1   WRITELN(" ",".":25,".":22,".":20,".":20,".":20);
01703  1   WRITE(" ","UTILIZATION":20,".":5,"  ");
01723  1   BINGO := 1;
01724  1   SYMBOL(UTIL1);
01726  1   BINGO := 2;
01727  1   WRITE(" ",".":25,"  ");
01742  1   SYMBOL(UTIL2);
01744  1
01744  1   FOR I := 1 TO 3 DO
01745  1   WRITELN(" ",".":25,".":22,".":20,".":20,".":20);
02000  1   WRITE(" ","ACQUISITIONS":20,".":5,"  ");
02020  1   BINGO := 1;
02021  1   SYMBOL(ACQUI1);
02023  1   BINGO := 2;
02024  1   WRITE(" ",".":25,"  ");
02037  1   SYMBOL(ACQUI2);
02041  1
02041  1   FOR I := 1 TO 3 DO
02042  1   WRITELN(" ",".":25,".":22,".":20,".":20,".":20);
02075  1   WRITE(" ","PRODUCTION":20,".":5,"  ");
02115  1   BINGO := 1;
02116  1   SYMBOL(PROD1);
02120  1   BINGO := 2;
02121  1   WRITE(" ",".":25,"  ");
02134  1   SYMBOL(PROD2);
02136  1
02136  1   FOR I := 1 TO 3 DO
02137  1   WRITELN(" ",".":25,".":22,".":20,".":20,".":20);
02172  1   WRITE(" ","EVALUATION":20,".":5,"  ");
02212  1   BINGO := 1;
02213  1   SYMBOL(EVAL1);
02215  1   BINGO := 2;
02216  1   WRITE(" ",".":25,"  ");
02231  1   SYMBOL(EVAL2);
02233  1
02233  1   FOR I := 1 TO 3 DO
02234  1   WRITELN(" ",".":25,".":22,".":20,".":20,".":20);
02267  1   WRITE(" ","ACTIVITY":20,".":5,"  ");
02307  1   BINGO := 1;
02310  1   SYMBOL(ACTIV1);
02312  1   BINGO := 2;
02313  1   WRITE(" ",".":25,"  ");
02326  1   SYMBOL(ACTIV2);
02330  1
02330  1   FOR SKIP := 1 TO 3 DO
02331  1   WRITELN(" ",".":25,".":22,".":20,".":20,".":20);
02364  1   WRITELN(" ",UNDER:67,UNDE:40);
02401  1   WRITELN;
```

```
02402  1   WRITELN(" ","3.0":27,"2.5":20,"2.0":20,"1.5":20,"1.0":20);
02436  1   WRITELN("-");
02441  1   WRITELN(" ","C O M P A R I S O N  O F  M E A N S":86);
02451  1   WRITELN("0","*** GROUP 1":61);
02461  1   WRITELN("0","XXX GROUP 2":61);
02471  1   (* WRITE OUT PERCENTAGES *)
02471  1   WRITELN("1");
02474  1   WRITELN(" "," ":25,"PERCENTAGES":46);
02510  1   WRITELN("0"," ":25,"GROUP  1":25,"GROUP  2":37);
02531  1   WRITELN("0"," ":25,"1       2       3":29,"1       2       3":38);
02552  1   WRITELN("0"," ":25,"P       P       P":29,"P       P       P":38);
02573  1   FOR I := 1 TO 50 DO
02574  1   BEGIN
02576  2   WRITELN(" "," ":25,I:9,".":1,PER[I,1]:7:0,PER[I,2]:7:0,PER[I,3]:7:0,
02647  2 * ":20,PER[I,4]:4:0,PER[I,5]:7:0,PER[I,6]:7:0);
02707  2   END;
02711  1 END.  (* PROGRAM ENDS HERE *)
```